Portable SOC Log Analyzer For Isolated Networks (Offline SIEM Lite)

Vishva G[1], Vijay R[2], Manivel K[3]
[1, 2, 3] Dept of CSE(Cyber Security)
[1, 2, 3] Sri Venkateswaraa College of Technology

Abstract- *The growing complexity of cyber threats has increased the importance of continuous monitoring and analysis of system logs. Analyzing system logs reveals patterns of activity that help uncover unusual behaviour and possible security issues. However, most modern Security Information and Event Management (SIEM) solutions rely on cloud infrastructure and require constant internet connectivity, which limits their usability in isolated or restricted environments.*

This paper introduces a Portable SOC Log Analyzer, a lightweight and offline-capable system designed to perform log analysis without external dependencies. The system collects logs from multiple sources, processes them into structured formats, and applies rule-based detection techniques to identify suspicious activities such as repeated login failures, unauthorized access attempts, and abnormal behaviour patterns.

The proposed solution includes features such as alert generation, log filtering, and graphical visualization through a user-friendly interface. Because it operates fully offline, the tool preserves privacy and remains dependable, which makes it well-suited for isolated or air-gapped systems.

Keywords: Air-Gapped Systems, Cybersecurity, Log Analysis, Offline SIEM, Threat Detection

I. INTRODUCTION

In recent years, the rapid expansion of digital technologies has significantly increased the risk of cyber-attacks. Every computing environment produces logs that capture user actions, internal processes, and network traffic, which can be examined for anomalies. These logs serve as an essential source for identifying security threats and diagnosing system issues.

In most organizations, SOC teams depend on SIEM platforms to continuously track logs and react quickly to security incidents. Despite their effectiveness, most SIEM tools require internet access, centralized servers, and significant financial investment. This makes them unsuitable for environments where connectivity is restricted or where resources are limited

Organizations operating in secure or isolated environments, such as research labs and defense systems, require tools that function independently of external networks. Similarly, small organizations and academic institutions need cost-effective alternatives that are easy to deploy and manage.

To address these challenges, this paper proposes a **Portable SOC Log Analyzer**, which provides essential log analysis and threat detection capabilities in an offline environment. The system is designed to be lightweight, portable, and user-friendly, making it accessible to a wide range of users.

II. IDENTIFY, RESEARCH AND COLLECT IDEA

The idea for this project originated from the limitations observed in existing SIEM systems. Most available solutions are either too complex or dependent on internet connectivity, which creates challenges for offline environments.

A detailed study of existing tools and research papers revealed the need for a simplified log analysis system that can operate independently. The goal was to design a solution that maintains essential security monitoring features while eliminating unnecessary complexity and external dependencies.

The research process involved analyzing different log formats, studying common attack patterns, and identifying key functionalities required for effective threat detection. Based on these observations, a portable and offline-capable log analysis system was conceptualized.

III. STUDIES AND FINDING

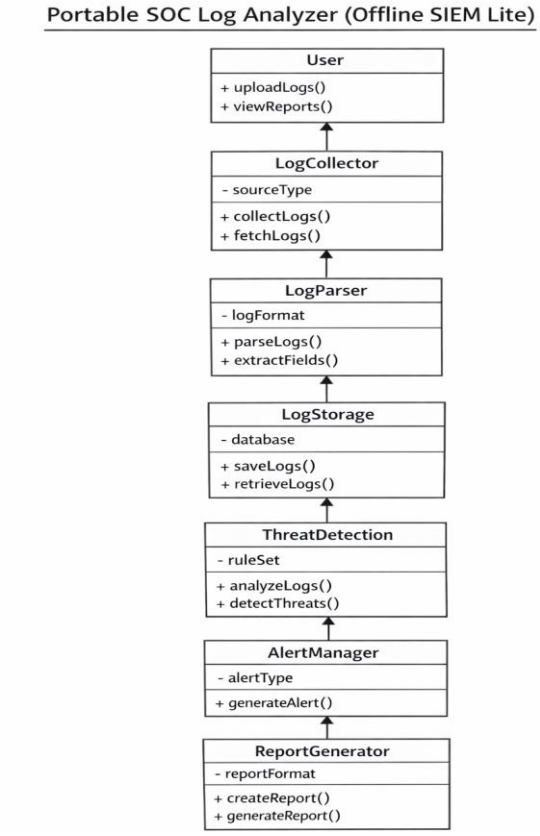
The proposed system is developed as a modular application that processes log data through multiple stages. Each module performs a specific function to ensure efficient log analysis and threat detection.

The system begins by collecting log files from various sources such as operating systems, servers, and applications. These logs are then processed using parsing techniques to extract relevant information such as timestamps, IP addresses, and event types. The processed data is stored in a lightweight database for further analysis.

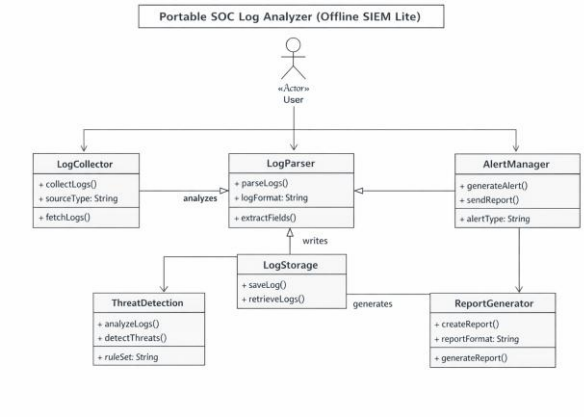
A rule-based detection engine is implemented to identify suspicious activities. For example, Login attempts are tracked, and the analyzer issues alerts if multiple failures occur within a short period, signaling potential brute force activity. Similarly, Connections originating from unfamiliar or untrusted IPs are highlighted as potentially risky.

The system also provides visualization features that present log data in graphical formats, making it easier for users to understand patterns and trends. Alerts are generated whenever suspicious activity is detected, allowing users to take appropriate action.

The findings indicate that the system is capable of performing efficient log analysis without relying on external resources. It provides a balance between functionality and simplicity, making it suitable for both learning and practical applications.



System Architecture



UML Diagram

IV. GET PEER REVIEW

Peer review plays a vital role in validating the quality, accuracy, and relevance of a research work before publication. In the context of the proposed Portable SOC Log Analyzer, the system design, implementation approach, and experimental outcomes were carefully evaluated through Feedback from peers helped refine the system so that it aligned with academic expectations while remaining practical for real-world use. Initially, the research paper and system prototype were shared with fellow students, faculty members, and individuals with basic knowledge of cybersecurity and software development. Reviewers were encouraged to examine multiple aspects of the project, including system architecture, clarity of methodology, correctness of detection logic, and overall presentation of the paper.

Special attention was given to the effectiveness of the rule-based detection mechanism. Reviewers analyzed whether the defined rules, such as identifying repeated login failures or unusual access patterns, were logically sound and capable of detecting real-world threats. Suggestions were provided to improve rule thresholds and include additional conditions to reduce false positives.

The usability of the system was another important evaluation factor. Feedback indicated that while the system was functional, improvements could be made in terms of interface clarity and navigation. Based on these inputs, the user interface was refined to ensure that even non-technical users could easily upload logs, interpret alerts, and analyze results.

Performance evaluation was also conducted during the peer review phase. The system was tested with varying sizes of log datasets to observe processing speed and stability. Reviewers suggested optimizing data handling techniques to

ensure consistent performance when dealing with large volumes of log data.

In addition, reviewers assessed the clarity and structure of the research paper itself. They provided feedback on improving the flow of content, enhancing technical explanations, and ensuring that all sections are well-connected and logically organized. This helped in refining the presentation and making the paper more suitable for publication.

Overall, the peer review process contributed significantly to improving both the technical implementation and the documentation of the project. It ensured that the system is reliable, user-friendly, and aligned with the objectives of providing an efficient offline log analysis solution.

V. IMPROVEMENT AS PER REVIEWER COMMENT

The feedback obtained during the peer review phase played a crucial role in refining both the system and the research documentation. Based on the suggestions provided by reviewers, several enhancements were systematically implemented to improve accuracy, usability, and overall performance of the Portable SOC Log Analyzer.

One of the primary areas of improvement was the **rule-based detection mechanism**. Reviewers pointed out that some detection rules were either too strict or too general, leading to false positives or missed alerts. To address this, the threshold values for events such as failed login attempts were carefully adjusted. Additional conditions were introduced to make the rules more context-aware, thereby improving the reliability of threat detection.

Another significant enhancement was made in the **log parsing and data processing module**. Initial implementation faced challenges in handling inconsistent log formats and noisy data. Based on reviewer feedback, the parsing logic was refined to improve data normalization and ensure accurate extraction of key fields such as timestamps, IP addresses, and event types. This resulted in better structured data and more efficient analysis.

The **user interface** was also improved considerably. Reviewers suggested simplifying navigation and enhancing the clarity of displayed information. In response, the dashboard layout was redesigned to present alerts, log summaries, and visualizations in a more intuitive manner. Clear labels, improved filtering options, and better organization of data were incorporated to enhance user experience.

Performance optimization was another key focus area. During testing, it was observed that processing large log files could affect system responsiveness. To overcome this, efficient data handling techniques were implemented, including optimized queries and reduced redundancy in data storage. These improvements ensured smoother performance even with larger datasets.

The **alert generation system** was further refined based on feedback. Initially, alerts lacked sufficient descriptive information, making it difficult for users to understand the context of the threat. Enhancements were made to include detailed descriptions, severity levels, and relevant log references, enabling users to take informed actions.

In addition to technical improvements, the **documentation of the research paper** was also revised. Reviewers recommended better structuring of content, clearer explanation of methodologies, and improved linkage between different sections. These suggestions were incorporated to enhance readability and ensure that the paper meets academic publication standards.

Overall, the incorporation of reviewer feedback significantly strengthened the system by improving its accuracy, efficiency, and usability. It also enhanced the quality of the research paper, making it more comprehensive and suitable for journal submission.

VI. CONCLUSION

The development of the **Portable SOC Log Analyzer (Offline SIEM Lite)** demonstrates an effective approach to performing log analysis and threat detection in environments where traditional SIEM systems are not feasible. Since the analyzer works without internet or heavy infrastructure, it offers a lightweight and practical option for environments with limited resources. The system successfully integrates essential functionalities such as log collection, parsing, analysis, alert generation, and visualization within a lightweight and portable framework. The implementation of rule-based detection techniques enables the identification of common security threats, including repeated login failures and unauthorized access attempts. Additionally, the user-friendly interface ensures that the system can be operated even by individuals with limited technical expertise.

The results obtained from testing indicate that the system performs efficiently in offline conditions, maintaining stability while processing large volumes of log data Because it

avoids heavy infrastructure costs, the analyzer is well-suited for smaller organizations, universities, and secure facilities where privacy must be maintained.

In summary, the proposed system achieves its objective of providing a simplified yet functional log analysis solution. It highlights the potential of lightweight cybersecurity tools in addressing real-world challenges and lays a strong foundation for further enhancements in intelligent threat detection and system scalability.

VII. APPENDIX

The appendix section provides additional supporting information related to the implementation of the **Portable SOC Log Analyzer (Offline SIEM Lite)**. It includes sample outputs, and brief descriptions of key components used in the project.

A. Sample Log Entry

Jan 12 10:15:32 server1 sshd[1234]: Failed password for invalid user admin from 192.168.1.10 port 22

This log entry represents a failed login attempt, which is detected by the system as a potential security threat based on predefined rules.

B. Sample Detection Rule

Example rule used in the system:

- **Rule Name**: Brute Force Detection
- **Condition**: Multiple failed login attempts
- **Threshold**: More than 5 attempts within a short duration
- **Severity**: High

C. Sample Alert Output

Alert ID: 101

Type: Brute Force Attack
Source IP: 192.168.1.10
Severity: High
Description: Multiple failed login attempts detected
Timestamp: 2026-01-12 10:20:00

This alert is generated when the system detects suspicious login activity.

D. Technologies Used

- Python (Core processing)
- Fast API / Flask (Backend framework)
- HTML, CSS, JavaScript (Frontend)
- SQLite (Database)

E. Key Functional Components

- Log Parsing Module
- Detection Engine
- Alert Generation System
- Visualization Dashboard
- Report Generation Module

VIII. ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the faculty members and mentors of the Department of Computer Science for their continuous guidance, encouragement, and valuable suggestions throughout the development of this project. Their support played a crucial role in shaping both the technical implementation and the research work.

We also extend our appreciation to our institution for providing the necessary resources and environment to carry out this work successfully. The availability of academic materials and technical support greatly contributed to the completion of this project.

Special thanks are due to our peers and colleagues who provided constructive feedback during the review phase, helping us improve the quality and effectiveness of the system. Their insights were instrumental in refining the design and enhancing the overall performance of the application.

Finally, we acknowledge all the researchers and authors whose work in the field of cybersecurity and log analysis served as an inspiration for this study.

REFERENCES

[1] W. Stallings, *Network Security Essentials: Applications and Standards*, 6th ed., Pearson Education, 2017.
[2] National Institute of Standards and Technology (NIST), “Guide to Intrusion Detection and Prevention Systems (IDPS),” Special Publication 800-94, 2012.
[3] K. Scarfone and P. Mell, “Guide to Intrusion Detection Systems,” NIST, 2007.
[4] R. Behl, *Cybersecurity and Cyberwar: What Everyone Needs to Know*, Oxford University Press, 2017.

[5] Splunk Documentation, “Security Information and Event Management Solutions,” Available: https://www.splunk.com
[6] IBM QRadar Documentation, “Security Intelligence Platform,” Available: https://www.ibm.com/security
[7] Elastic Stack Documentation, “Log Analysis and Monitoring Tools,” Available: https://www.elastic.co
[8] S. Axelsson, “Intrusion Detection Systems: A Survey and Taxonomy,” Technical Report, Chalmers University, 2000.
[9] M. Bishop, *Computer Security: Art and Science*, Addison-Wesley, 2003.
[10] Research articles on log analysis, anomaly detection, and SIEM systems from IEEE and ACM digital library.